33270

H04L

(12) UK Patent Application (19) GB (11) 2 350 985 (13) A

(43) Date of A Publication 13.12.2000

(21) Application No 9913676.4

(22) Date of Filing 11.06.1999

(51) INT CL[7]
H04L 7/10 , G06F 13/37

(52) UK CL (Edition R)
H4P PSB

(56) Documents Cited
US 5574951 A US 5423053 A US 4937815 A

(58) Field of Search
UK CL (Edition Q) H4P PSB PSEX PSX
INT CL[6] G06F , H04L

(71) Applicant(s)
Mitel Corporation
(Incorporated in Canada - Ontario)
350 Legget Drive, Kanata, Ontario, K2K 1X3, Canada

(72) Inventor(s)
Jerzy Wieczorkiewicz
Andrew Lawrence Yeaton

(74) Agent and/or Address for Service
Marks & Clerk
4220 Nash Court, Oxford Business Park South,
OXFORD, OX4 2RU, United Kingdom

(54) Abstract Title
System architecture

(57) A system architecture includes a central controller 10 having a framing pulse output FP, a serial digital bus 12 connected to the central controller, and a plurality of peripheral devices 20 connected to the serial bus. Each peripheral device has a framing pulse input FP and a delayed framing pulse output FPD. The peripheral devices and the central controller are connected in a daisy chain arrangement so that the framing pulse output of the central controller is connected to the framing pulse input of the first device. The delayed framing pulse output of the first device is connected to the framing pulse input of the second device and so on throughout the chain. In this way the individual peripheral devices can be addressed in a frame defined by their framing pulse input and delayed framing pulse output.

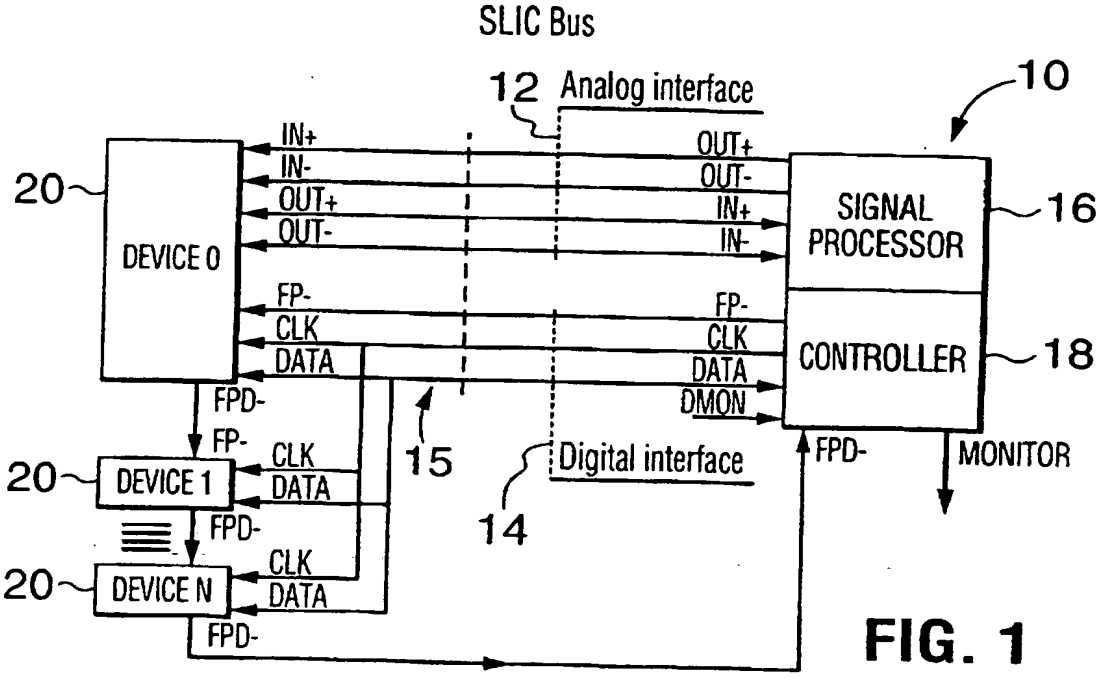

FIG. 1

At least one drawing originally filed was informal and the print reproduced here is taken from a later filed formal copy.

This print takes account of replacement documents submitted after the date of filing to enable the application to comply with the formal requirements of the Patents Rules 1995

GB 2 350 985 A

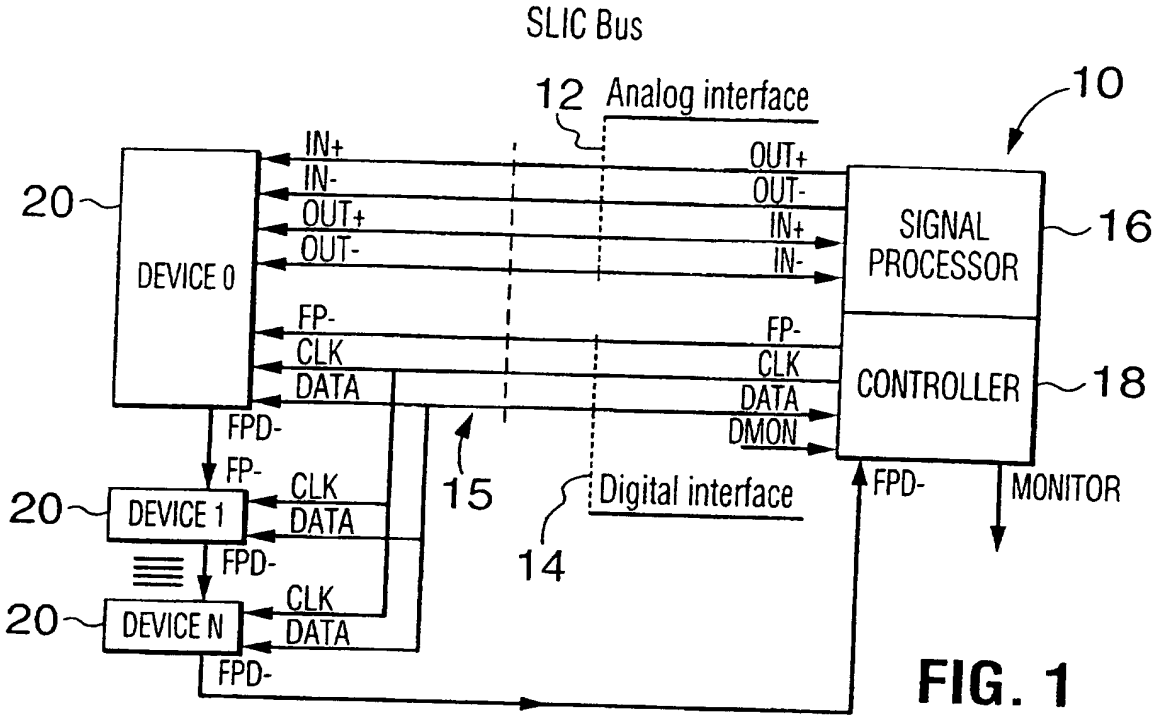

FIG. 1

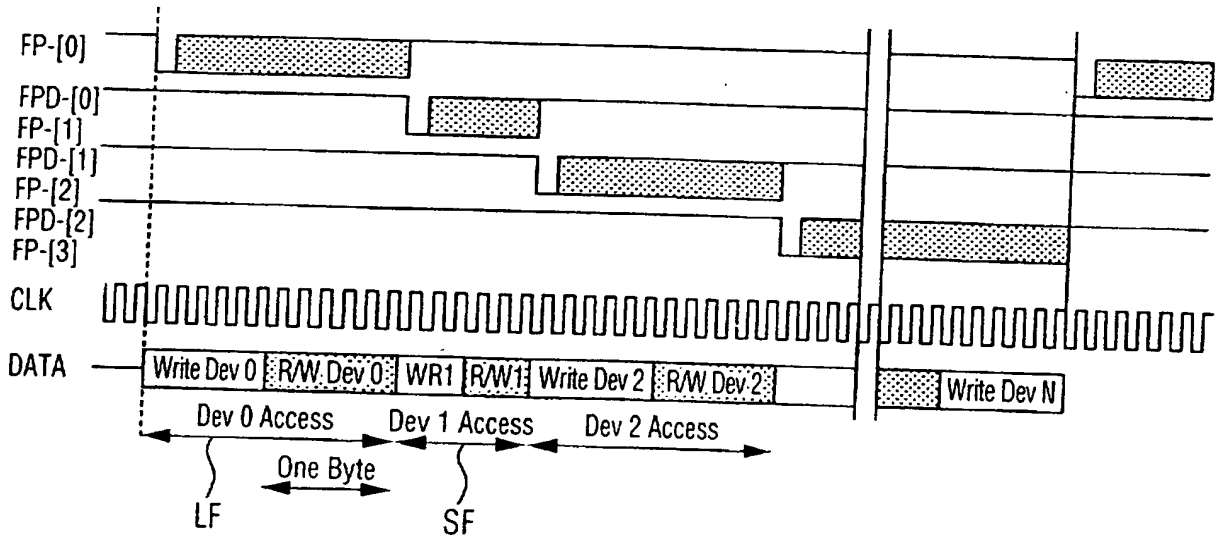

FIG. 2

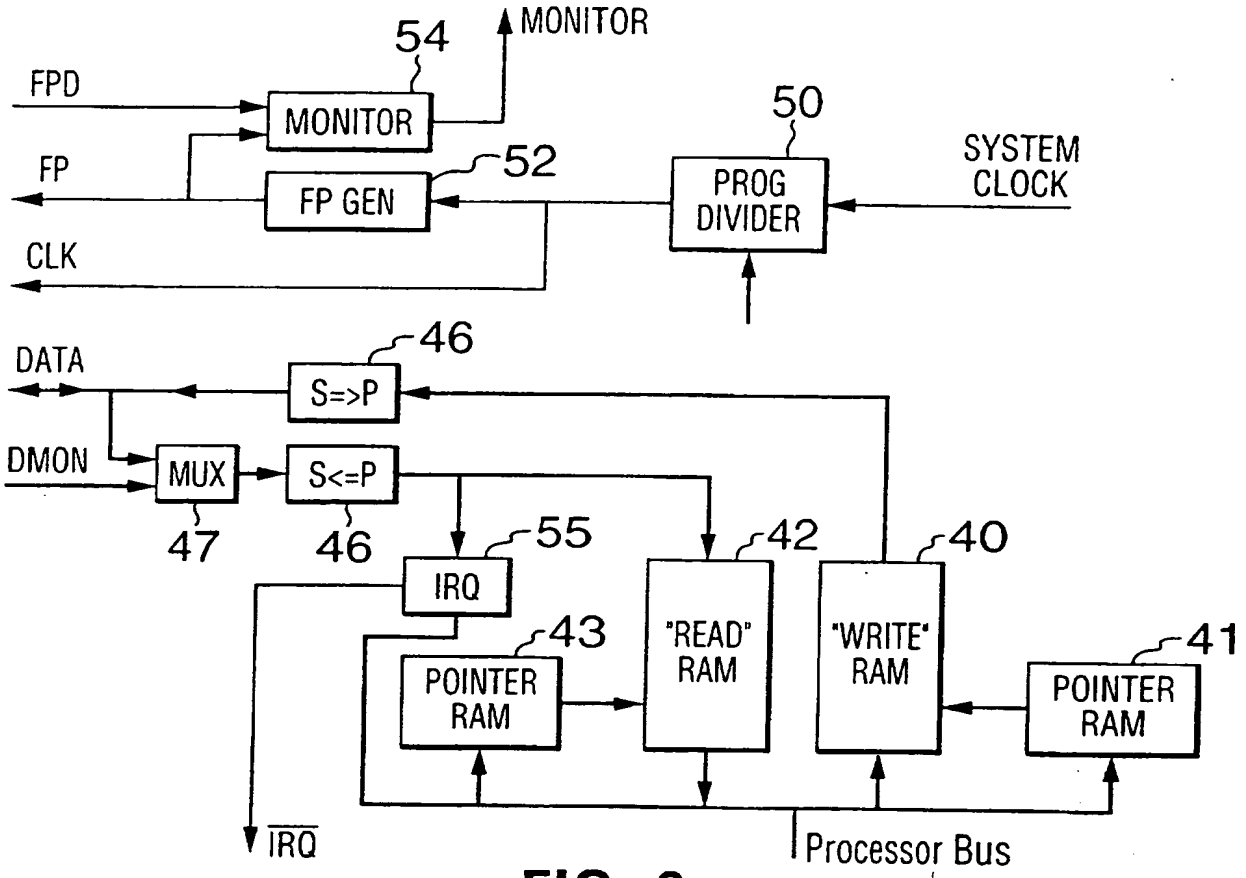

FIG. 3

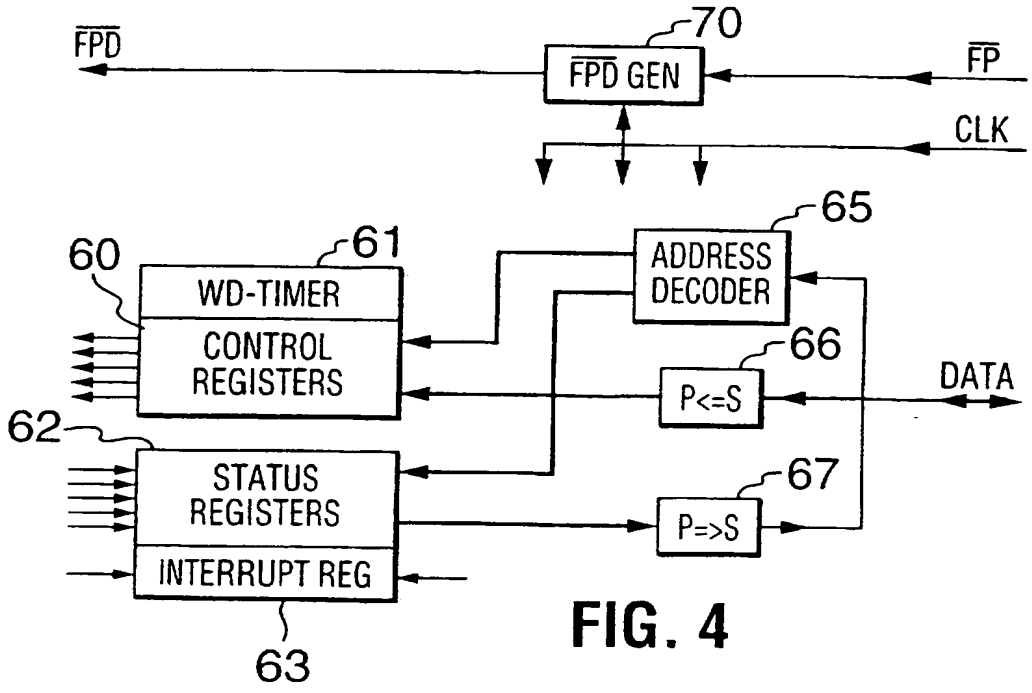

FIG. 4

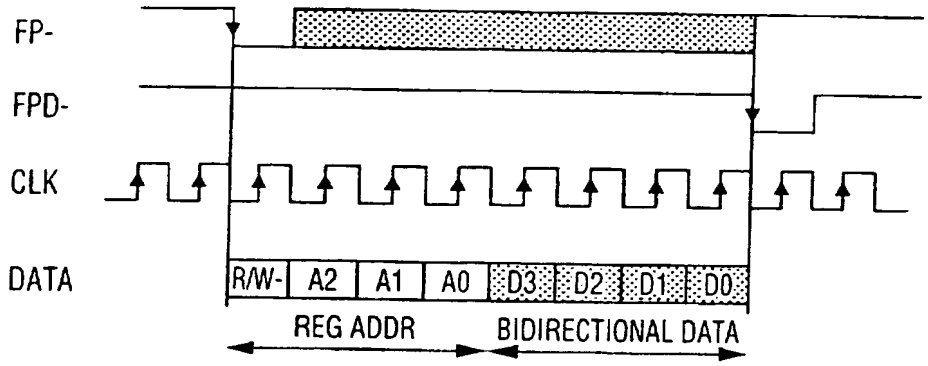

FIG. 5

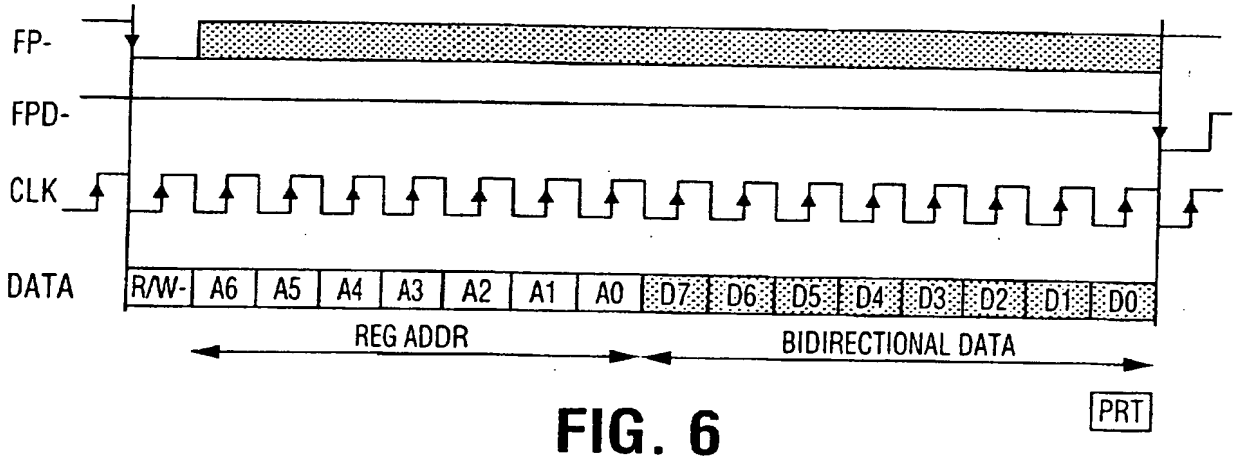

FIG. 6

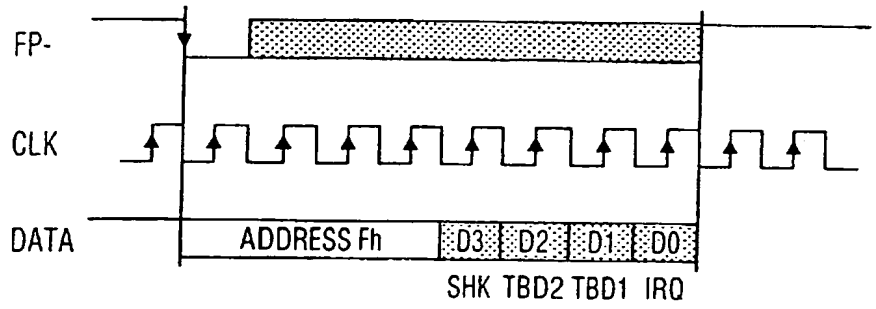

FIG. 7

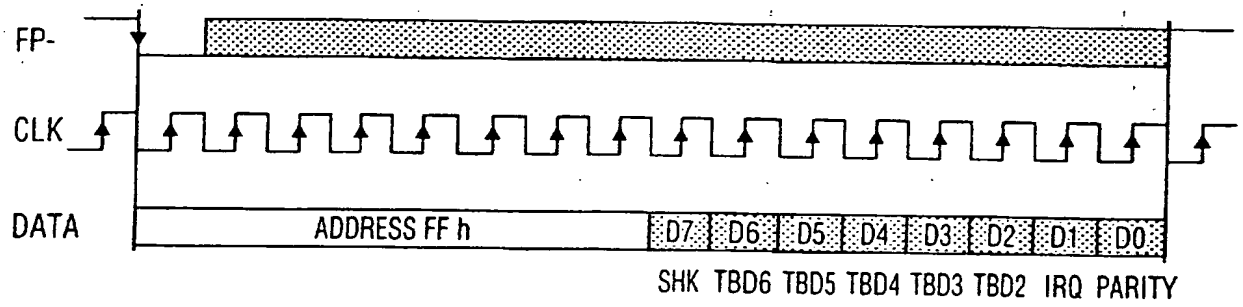

FIG. 8

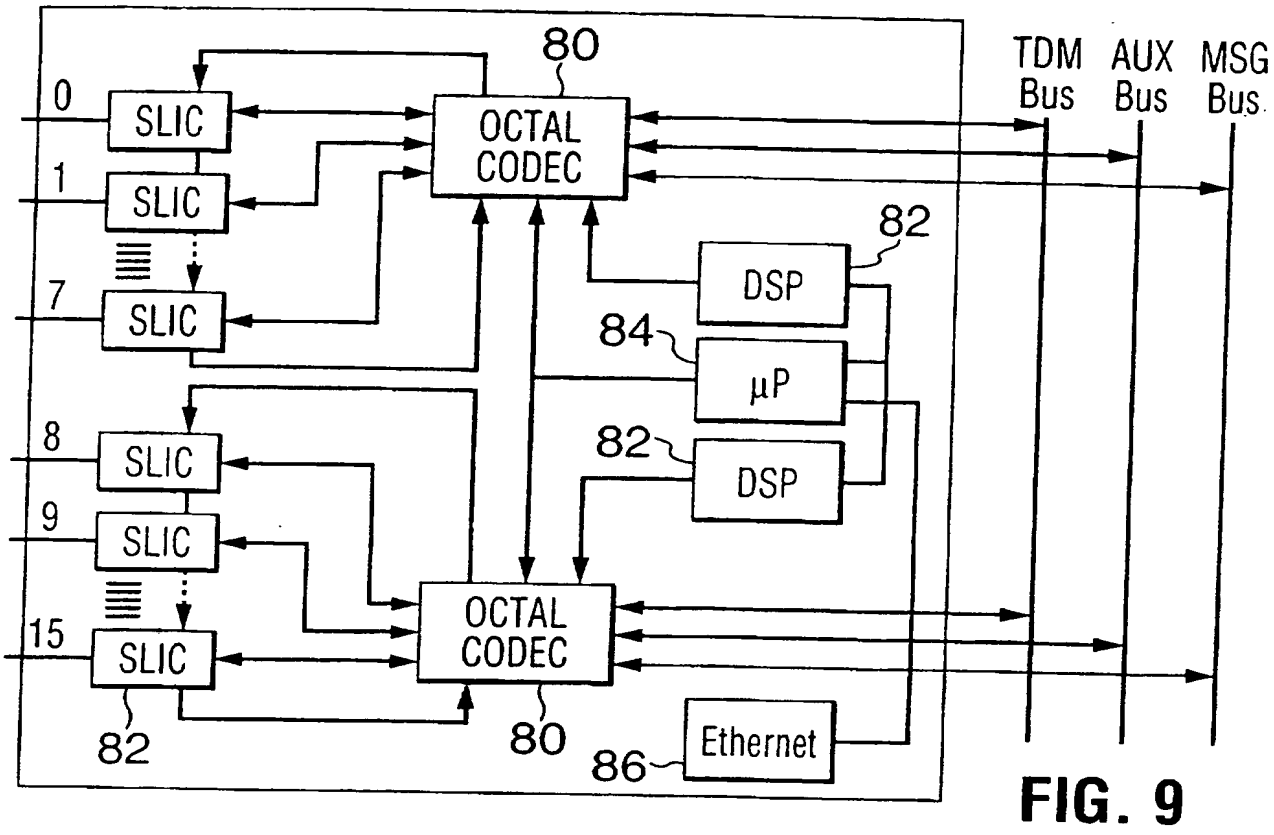

FIG. 9

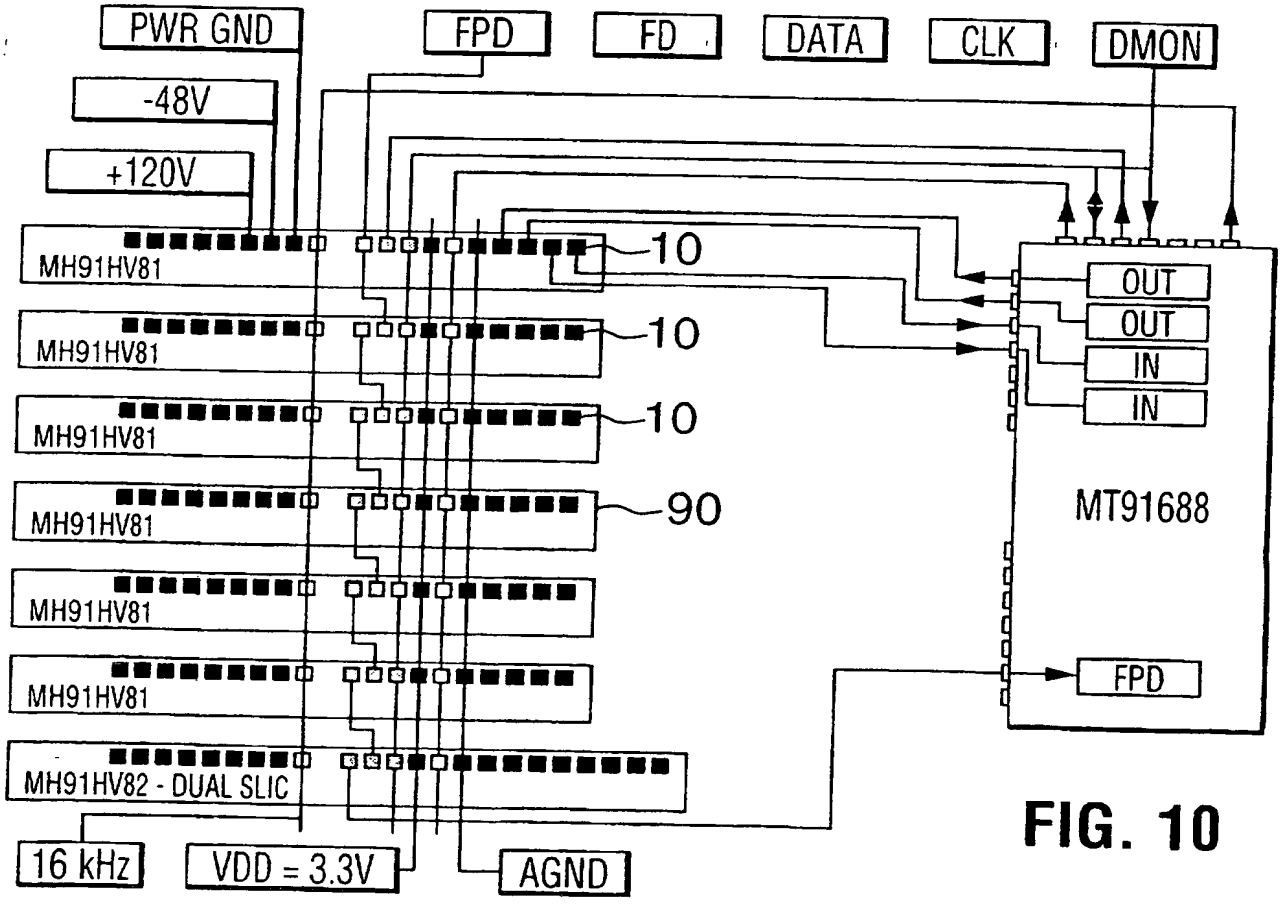

FIG. 10

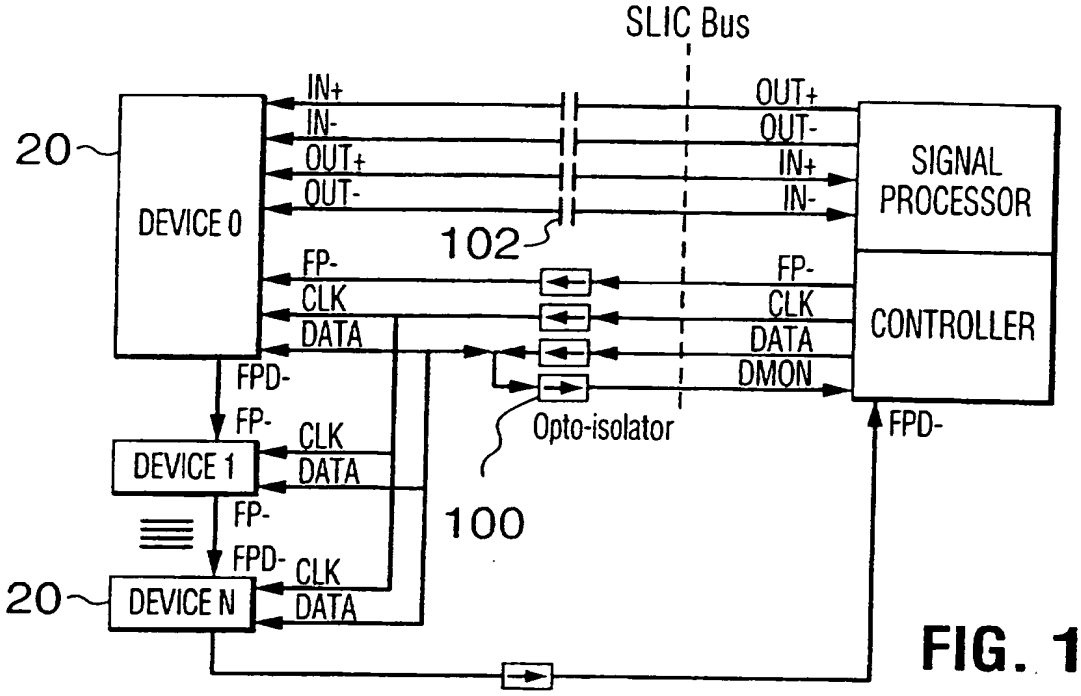

FIG. 11

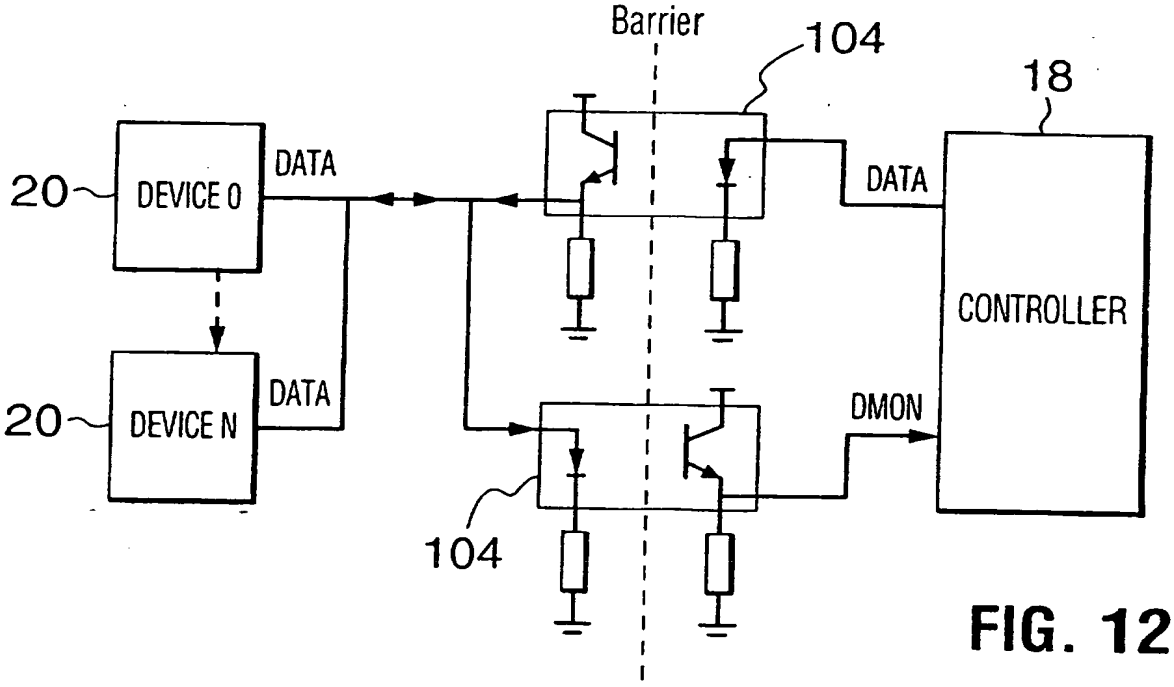

FIG. 12

SYSTEM ARCHITECTURE FOR ELECTRONIC DEVICES

This invention relates to a system architecture for connecting multiple peripheral devices to a single controller, and particularly but not exclusively to a system comprising multiple complex devices connected to a single controller in a telecommunications line card.

In order to connected peripheral devices, such as line cards, to a central controller, it necessary to provide a high pin count on integrated circuits, taking up scarce chip real estate, and route numerous wires to the various pins, thereby increasing the complexity of the system.

An object of the invention is to provide a system that permits complex peripheral devices to be connected to a central controller in an efficient and simple manner.

According to the present invention there is provided a system architecture comprising a central controller having a framing pulse output, a serial digital bus connected to said central controller, a plurality of peripheral devices connected to said serial bus, and each said peripheral device having a framing pulse input and a delayed framing pulse output, said peripheral devices and said central controller being connected in a daisy chain arrangement so that the framing pulse output of said central controller is connected to the framing pulse input of the first device, the delayed framing pulse output of the first device is connected to the framing pulse input of the second device and so on throughout the chain, whereby said individual peripheral devices can be addressed in a frame determined by their framing pulse input and delayed framing pulse output.

The serial bus, which is preferably two-wire, with daisy chain addressing permits bi-directional transfer on a single data wire with more than one kind of data format, for example, long frames and short frames. The architecture is extremely flexible since there is no inherent limitation on the number of devices connected except for bus capacitance.

The last peripheral device in the chain preferably inputs its delayed framing pulse to a delayed framing pulse input of the controller to permit monitoring of bus integrity. Flexible clocking can be employed.

In addition, the digital bus can be associated with a fully differential analog bus to permit non-overlapping routing of signals on printed circuit boards. The fully differential bus with differential input and differential output provides increased signal-to-noise

performance. The analog bus preferably has a high impedance on its analog outputs and provides isolated ground support if required.

The device can easily be made compatible with 2.7V to 3.6V and 4.75V to 5.25V power supplies.

The primary application of the device is in the field of interconnecting multiple voice-processing devices (e.g. connecting Subscriber Line Interface Circuits and Central Office Line Interface Circuits to Voice Codecs). However, the simplicity and flexibility of this bus extend its usage beyond telecom or voice processing applications.

The invention also provides a method of establishing communication between a plurality of peripheral devices and a central controller, comprising the steps of connecting said devices to a serial bus in a daisy chain arrangement, sending a framing pulse from a central controller to a first of said peripheral devices; defining a framing period in said first peripheral device based on said framing pulse; generating a delayed framing pulse in said first peripheral device and sending said delayed framing pulse to a framing pulse input of a second said peripheral device; and defining a framing period in said second peripheral device based on said delayed framing pulse generated in the previous peripheral device and so on along the chain of said peripheral devices.

The invention will now be described in more detail, by way of example only, with reference to the accompanying drawings, in which:-

Figure 1 shows the architecture of a serial bus in accordance with the invention;

Figure 2 shows the functional timing for the digital interface;

Figure 3 is a functional block diagram of the bus controller;

Figure 4 is a functional block diagram of a bus peripheral;

Figure 5 shows the read/write operation in the short frame mode;

Figure 6 shows the read/write operations in the long frame mode;

Figure 7 shows the status register timing in the short frame mode;

Figure 8 shows the status register timing in the long frame mode;

Figure 9 shows the architecture of a line card including bus controllers;

Figure 10 shows a printed circuit board layout including an interface device;

Figure 11 shows in interface in an isolated ground application; and

Figure 12 shows an implementation of a bi-directional data connection.

Referring now to Figure 1, the bus interface device 10 has two distinct interfaces, namely an analog interface 12 and digital interface 14. The analog interface 12 provides a uniform connection between analog ports on peripheral devices 20 and signal processor 16. The digital interface 14 originates in the controller 18, which controls and monitors the attached peripheral devices 20. The analog interface 12 is fully differential and includes differential in and differential out lines. The digital interface 14 includes a clock line CLK, a frame pulse line FP, and a data line DATA.

Each peripheral device 20 has CLK and DATA ports connected to the respective CLK and DATA lines of digital bus 15. In addition the peripheral devices $20^0 \ldots 20^N$ have input ports FP for receiving framing pulses and output ports FPD for outputting delayed framing pulses. The peripheral devices $20^0 \ldots 20^N$ are arranged in a daisy chain fashion so that first framing pulse FP from the controller 18 is applied to the FP input of the first peripheral device 20^0, the delayed framing pulse output FPD from the first peripheral device is applied to the FP input of the device 20^1, and so on. The delayed framing pulse FPD from the last peripheral device 20^N is applied to the delayed framing pulse input FPD of the controller 18. This permits the controller 18 to monitor the integrity of the bus 15. More generally, the FPD output of a peripheral device 20^n is connected to the FP input of peripheral device 20^{n+1}.

As noted above, the devices connected to the bus 15 are arranged in daisy chain. This chain is created by connecting delayed frame pulse output (FPD-) of one device to the frame pulse input (FP-) of the following device. The originating frame pulse FP- is generated by the bus controller 18. Each device in the chain can be individually addressed if its physical location in the chain is known by selecting its associated framing pulse.

The functional timing diagram is shown in Fig 2. As will be seen in the Figure, peripheral device 20^0 is addressed during framing pulse FP[0], device 20^1 is addressed during framing pulse FP[1], and so on.

The controller 18 continuously checks integrity of framing by counting number of clock cycles occurring between frame pulse sent out on the FP port of the controller 18 and received at the delayed frame pulse port FPD and generated by the last device in a chain. The error in integrity check is used to generate major alarm in the system when the number of clock cycles in inconsistent with the known number of devices in the chain.

The controller 18 is shown in more detail in Figure 3. It comprises a write RAM 40 associated with a pointer RAM 41, and a read RAM 42 associated with a pointer RAM 43. The write RAM 40 is connected through serial parallel-to-serial converter 46 to the data line of bus 15. Incoming data is fed through mux 47 and serial-to-parallel converter 46 to read RAM 42. IRQ generator 55 generates interrupts.

Programmable divider 50 takes the system clock and divides it by a programmable coefficient to generate the CLK output for CLK line on the bus. The CLK pulse is also input to framing pulse generator 52, which generates the FP output from the controller 18. This is also fed to the input of the integrity monitor 54, which receives at its other input the delayed framing pulse FPD from the last peripheral device in the chain.

The CLK clock is used internally to determines the time of transfer of data onto the bus 15. The monitor 54 continuously monitors integrity of the bus by comparing the received delayed frame pulse FPD from the last device in a chain with its own knowledge of when FPD- should arrive based on how may devices are connected to the bus 15. If this verification fails, the controller will generate an interrupt, pull the MONITOR pin low and trigger the major fault alarm.

All the data exchanged between the controller 18 and the peripheral devices is stored in the two memories 40, 42. Allocation of memory space for individual devices is controlled by the associated pointer RAM 41, 43. Flexible pointers allow the controller 18 to service simple and complex peripheral devices 20 with efficient use of resources.

The block diagram of the peripheral device interface is shown in Figure 4. The peripheral device interface consists of control registers 60 with associated watchdog timer 61 and status registers 62 with associated interrupt register 63. The bi-directional DATA line of the bus is connected to serial-to-parallel converter 66, parallel-to-serial converter 67, and

address decoder 65. Incoming framing pulses are connected to delayed framing pulse generator 70.

The peripheral devices 30 take the CLK clock and FP- frame pulse as input and generate from them a delayed frame pulse FPD-. The delay between FP- and FPD- is determined by design. Depending on the complexity of the peripheral device this delay can be 8 clock cycles for Short Data devices and 16 clock cycles for the Long Data devices discussed below.

Information sent from the bus controller 18 enters the DATA input of the peripheral device 20 where it is split into an address and a data field. Destination of the data is determined by the state of the R/W- bit. When the R/W bit is low the data is shifted from DATA line into serial-to-parallel register. After completion of the access cycle to the peripheral device (end of FPD-) the data from the serial-to-parallel register is transferred to the selected Control Register 60. When the R/W bit is high, the data from the selected Status or Interrupt register is transferred into parallel-to-serial register and shifted out on the DATA line.

The architecture supports two types of data structure. The Short Data type (four bits long) is intended for simple peripheral devices that do not require complex control. The Long Data type (eight bits long) is supporting complex devices that generate or receive numerical data over the bus 15. Devices supporting short or long type of data can be mixed in any combination on the bus.

The bus architecture does not limit how many devices can be connected to the bus. The only limiting factors are driving capability of individual devices in the chain and the addressing capacity of bus controller 18. As an example, a peripheral card may have a bus controller that supports up to twelve devices (eight mandatory devices plus four optional). The peripheral devices 20 can work with a minimum of 32 (thirty-one peripherals plus one controller) other devices connected to the bus (located on the same printed circuit board).

As also shown in more detail Figure 5, the falling edge of the frame pulse FP- selects the device for Read/Write (R/W) operations and triggers a delayed frame pulse generation circuit . This circuit counts eight or sixteen clock cycles and generates a delayed frame

pulse FPD-. During the time from the failing edge of FP- to the falling edge of the delayed frame pulse FPD- the device is accessible to the bus controller 18.

The falling edge of the frame pulse is created from the falling edge of the clock. The framing length, which is an inherent feature of a peripheral device, determines over how many bits the peripheral device will be active and when the delayed frame pulse will be generated.

The short and long data structures are also supported by two types of frame, short frames SF, and long frames LF depending on the amount of data to be transferred to the peripheral device. In Figure 2, device 20^0 has a long frame and device 20^1 has a short frame SF, but it will be understood that the configuration of long and short frames can be chosen in accordance with the device needs.

As shown in Figure 5, the short frame stretches over eight clock cycles, i.e. eight bits. The first bit transmitted is the read/write (R/W) bit. This is followed by three bits of the device's internal register address. The direction of these first four bits is always from controller to the peripheral device. These bits are followed by four bits representing a nibble of data that can flow in ether directions depending on state of the R/W- bit. The minimum frame pulse width is equal to one clock cycle but can be stretched up to the full access time of a device (8 bits).

The Read/Write (R/W) bit only defines direction of data transfer in the second nibble, i.e. the nibble of data. When this bit is set to high, the data nibble will be read from the peripheral device. When the R/W- bit is set to low the data nibble will be written to the peripheral device in the example shown.

Address bits A2 to A0 identify the register that is to be accessed during read or write operations.

Bits D3 to DO represent the data nibble permitting bi-directional transfer depending whether the registers are set for read or write operations.

The long framing pulse, shown in Figure 6, is intended for applications that require sophisticated control or transfer of numerical data. The frame structure is the similar to the short framing structure except for the number of bits transferred. In the long frame mode, sixteen bits are transferred during a single access cycle. The first bit sent is the

R/W- bit, followed by seven address bits. The allocated address space allows to access up to 128 registers inside a single peripheral device. Following the address is eight bits of data. Depending on the state of the R/W- bit in the first byte the data-will be written to or read from the device.

The minimum frame pulse width is equal to one clock cycle but can be stretched up to the full access time of the device (16 bits).

In the long frame structure, the Read/Write bit defines direction of data transfer in the second byte. When this bit is set to high the data byte will be read from the peripheral device. When the R/W- is set to low the data byte will be written into the peripheral device.

Address bits A6 to AO identify registers to be accessed during read or write operations.

Data bits D7 to DO permit bi-directional transfer of bytes of data.

A parity check for error detection can be performed on all fifteen bits transferred and the result of Even Parity check placed in the last bit sent. The use of a parity bit is optional.

The bus architecture supports two types of clocking arrangements that depend on the use of a Watchdog Timer (WT). The watchdog timer is a mechanism that is used to determine whether the system is functioning properly.

Peripheral devices with Watchdog Timers are designed to operate with the following clock frequencies: 32 kHz, 64 kHz, 96kHz (= 8 * 12 kHz = 6 * 16 kHz), 128 kHz, 256 kHz, 512 kHz, 1024 kHz, 1544 kHz and 2048 kHz. Selection of the clock frequency is software programmable. The embedded watchdog timer uses an on-chip PLL as its reference clock. This PLL is locked to the CLK clock during normal operation and falls into free-run mode when CLIK clock fails.

Not all peripheral devices need include a watchdog timer 61. Those without must operate with any clock in the range of 32 kHz to 2048 kHz. However, it is recommended that for optimum performance of the analog circuitry on the Bus (e.g. sigma-delta converters) the CLK clock will be frequency locked to the system clock.

The number of registers that can be addressed inside the peripheral device depends on the type of framing being implemented by the device. With Short Framing, the three address bits allow to address up to 8 nibbles or a total of 32 bits (read 32 bits and write 32 bits).

The Long Framing has seven address bits and they allow to access up to 128 read and 128 write registers (bytes) within a single device. This wide address space makes it possible to design each register to be of write/read type. Users can write data to a register and then read it back for checking. The Interrupt and Status Registers are not exceptions. They can be written for testing or reset purposes.

Most of the time there is very little activity on the bus. In such an idle period of time, the controller sends an Address Byte set to Fh (for SF) and FFh (for LF). This address points to the device status register that contains status and interrupt information as shown in Figures 7 and 8.

The status register 62 has only four or eight bits (depending on framing type) capacity that may not be insufficient to represent all-important events triggered inside a peripheral device 20. In such a case some bits in this register can be assigned as "collector" bits that represent the state of other interrupt registers in a device. Each of these bits will point to a distinctive Interrupt Register inside the device.

In the Long Framing mode the least significant bit may be used as parity check.

The switch Hook State bit is not an interrupt. The state of this bit always represents the present state of SHK.

Returning to Figure 1, the analog interface 12 consists of a two-wire differential input and a two-wire differential output. The input can be connected directly to the output. Capacitive coupling is only required in applications where grounds are not galvanically connected.

The analog bus defines 0 dBm0 signal level measured differentially as equivalent to 1 Vrms. This signal level guarantees operation over extended power supply variations starting from 2.7V.

The differential configuration provides for improved signal to noise performance and increased dynamic range. The following signal levels are typically selected as reference (differential measurements):

A-law 0.00 dBm - equivalent voltage level equal to 1.066 Vrms

3.14 dBm - equivalent voltage level equal to 1.530 Vrms (4.33 Vpp)

μ-law 0.00 dBm equivalent voltage level equal to 1.026 Vrms

3.17 dBm equivalent voltage level equal to 1.478 Vrms (4.18 Vpp)

The maximum load on the output driver should not exceed 100pF in parallel with 10 kohm. The output driver can be forced into high impedance state to allow multiplexing of analog signals. Drivers are biased to Vdd/2.

The differential inputs are compatible with the differential outputs. The inputs are self biased to Vdd/2 and have AC impedance not lower that 20 M.

Telephone line cards are typical applications for the device. The line card shown in Figure 9 comprises sixteen interface devices connected to OCTAL codes 80, which in turn are connected to digital signal processors 82 and microprocessor 84. Microprocessor 84 is connected to Ethernet card 86. However, the interface device can be used in applications outside telecommunications.

Figure 10 shows a typical printed circuit board (PCB) layout. Multiple interface devices 10 are connected to multichannel codecs 90. It will be seen that the layout is greatly simplified.

The bus may be used in applications where peripheral devices 20 and controller 18 are not galvanically connected, as is shown in Fig 11. In such a case, analog signals can be isolated through high voltage capacitors 102 and digital interface signals can be isolated by through optoisolators 100. The use of transformers in place of the capacitors 102 is also possible.

As shown in Figure 12, the DATA wire in the interface requires an optical splitter 104 to allow for a bi-directional flow of signals. The composite DATA signal, for the Read as well as for the Write operation appears on the monitor input DMON of the controller 18.

It is also possible to provide bi-directional signals on the analog bus.

It will be seen that the described interface device and bus provides an efficient and simple technique for establishing communication between a central controller and peripheral devices.

17. A method as claimed in any one of claims 10 to 16, wherein said peripheral devices are further interconnected by an analog bus.

18. A method as claimed in claim 17, wherein said analog bus supports fully differential inputs and outputs.

19. A bus controller for a serial bus for connecting peripheral devices in a daisy chain to a central controller, comprising a serial-to-parallel converter for receiving data from a serial data line forming part of the bus, a parallel-to-serial converter for sending data onto said bus, a read memory connected to said serial-to-parallel converter for storing data received from said bus, a write memory connected to said parallel-to-serial converter for storing data to be sent to the peripheral devices over the bus, pointer memories for identifying memory locations in said read and write memories where the data is to be stored, a clock pulse generator for sending clock pulses onto a clock line forming part of said bus, frame pulse generator for sending frame pulses out on a framing pulse line to a first of the peripheral devices to determine when the peripheral devices are available for addressing.

20. A bus controller as claimed in claim 19, further comprising a delayed framing pulse input for receiving delayed framing pulses from a last peripheral device in the daisy chain, and a monitor device for verifying the number of interim clock cycles.

21. A bus controller as claimed in claim 20, further comprising a multiplexer connected to the input of said serial-to-parallel converter, the data line being applied to one input of said multiplexer.

Application No: GB 9913676.4 **Examiner:** Al Strayton
Claims searched: 1-18 **Date of search:** 24 November 1999

Patents Act 1977
Search Report under Section 17

Databases searched:

UK Patent Office collections, including GB, EP, WO & US patent specifications, in:

UK Cl (Ed.Q): H4P: PSB; PSEX; PSX

Int Cl (Ed.6): G06F, H04L

Other:

Documents considered to be relevant:

Category	Identity of document and relevant passage		Relevant to claims
A	US 5 574 951	(SAWYER...)	
A	US 5 423 053	(CAHEN)	
A	US 4 937 815	(LIGHTHART)	

X	Document indicating lack of novelty or inventive step	A	Document indicating technological background and/or state of the art.
Y	Document indicating lack of inventive step if combined with one or more other documents of same category.	P	Document published on or after the declared priority date but before the filing date of this invention.
&	Member of the same patent family	E	Patent document published on or after, but with priority date earlier than, the filing date of this application.

BNSDOCID: <GB_____2350985A__I_>